

~~31.~~ 31. (Amended) A data processing system, comprising:

a remote control unit; and

a base unit connectable to a monitor for providing Internet access under the control of the remote control unit, the base unit comprising:

a processor; and

a memory electrically connected to the processor, the memory having stored therein Internet protocol [means] instructions [;] and [a client component of] a file transfer download routine[, the client component having an associated server component supported on a server; wherein the client component of the file transfer download routine is run by the processor and includes means] to be executed by the processor for performing the following steps:

receiving from a server a profile of a download file that includes identifying information for a plurality of component files that make up the download file;

initiating a download sequence by which each component file is transferred, one-by-one, from the server using the Internet protocol instructions;
and

responsive to receipt of the component [pieces] files [of a file] for reassembling the component [pieces] files into the download file using [a file] the profile.

REMARKS

Claims 1-10, 12, 14-17, 21-25, and 31 are pending in the present application.

Claims 11, 13, 18-20, and 26-30 are canceled. Claims 1-3, 5, 7-9, 12, 14, 15, 17, 21-25, and 31 are amended. Reconsideration of the claims is respectfully requested.

I. 35 U.S.C. § 102, Anticipation

The examiner has rejected claims 1-4, 8, 17, 21-25, and 31 under 35 U.S.C. § 102 as being anticipated by Averbuch et al. (US Patent No. 5,689,825), hereinafter referred to

as "*Averbuch*." The Office Action states that the claims are rejected as stated in prior Office Action (paper #6). This rejection is respectfully traversed.

Applicant's claim 1, as amended, recites:

1. A method of downloading a download file, consisting of a set of component files, from an Internet server, comprising:
 - receiving from the server a profile of the download file that includes identifying information for each component file;
 - initiating a download sequence by which each component file is transferred, one-by-one, from the server using an Internet protocol;
 - when the download sequence is complete, reassembling the component files into the download file using the identifying information in the profile.

Averbuch does not teach or fairly suggest a method of downloading a download file comprising "receiving from the server a profile of the download file that includes identifying information for each component file," as recited, in combination, in amended claim 1. According to an exemplary embodiment of the present invention, a large file to be transferred from a server to a client is broken into a plurality of component files at the server. It is the component files that are individually identified and downloaded by the client and reassembled to form the original download file. These features are supported by the specification at least at page 17, line 12, to page 19, line 22.

The Office Action states:

Col.5 lines 36-39, *Averbuch* discloses the server inform (i.e. generating) the battery charger the size of the updated software to be downloaded (i.e. the claimed profile). See col.5 line 38.

Office Action dated 18 July 2000. Applicant respectfully disagrees. *Averbuch* teaches a method of downloading a file from a server to a client comprising the server transferring information including only the size of the file to the client. The client then transfers download session parameters, including a block size and a number of blocks remaining to be transferred, to the server. See *Averbuch*, col. 5, lines 28-55. However, the blocks of *Averbuch* are not "component files," as recited in amended claim 1. The client does not receive from the server "a profile of the download file that includes identifying information for each component file," because the blocks of *Averbuch* are not individually predetermined and **identified** by the server. Therefore, *Averbuch* does not

teach or suggest the “profile” of the presently claimed invention. It follows that *Averbuch* does not teach or suggest “reassembling the component files into the download file **using the identifying information in the profile,**” because *Averbuch* does not teach or suggest identifying information in a profile.

Furthermore, *Averbuch* does not teach, suggest, or give any incentive to make the needed changes to reach the presently claimed invention. *Averbuch* actually teaches away from the presently claimed invention because it teaches determining block size and, thus, the number of blocks at download time opposed to the server identifying a plurality of predetermined component files as in the presently claimed invention. Absent the examiner pointing out some teaching or inventive to implement *Averbuch* with predetermined component files at the server, one of ordinary skill in the art would not be led to modify *Averbuch* to reach the present invention when the reference is examined as a whole. Absent some teaching, suggestion, or incentive to modify *Averbuch* in this manner, the presently claimed invention can be reached only through an improper use of hindsight using the applicants’ disclosure as a template to make the necessary changes to reach the claimed invention.

Since claims 2-4 depend from claim 1, the same distinctions between *Averbuch* and the claimed invention in claim 1 apply for these claims. Additionally, claims 2-4 claim other additional combinations of features not suggested by the reference. Consequently, it is respectfully urged that the rejection of claims 1-4 is overcome.

Applicant’s claim 8 recites:

8. A method of downloading a download file, consisting of a set of component files, to an Internet client, comprising:
 - breaking the download file into the set of component files;
 - generating a profile of the download file that includes identifying information for each component file;
 - initiating a download sequence by which each component file is transferred, one-by-one, to the client using an Internet protocol; and
 - responsive to any interruption of the download sequence, restarting the download sequence with a component file affected by the interruption.

Averbuch does not teach or fairly suggest a method of downloading a download file comprising “breaking the download file into a set of component files” and “generating a

profile of the download file that includes identifying information for each component file,” as recited, in combination, in amended claim 8. According to an exemplary embodiment of the present invention, the server code breaks the file into component files, which may be stored in addressable locations. This feature is supported by the specification at least at page 19, lines 4-22.

In contradistinction, *Averbuch* simply teaches that blocks are transferred from the server to the client. See *Averbuch*, col. 5, lines 56-62. *Averbuch* does not teach breaking the download file into a set of component files. The transfer of blocks in *Averbuch* is merely the breaking of the transmission of the file into portions. Each block is not itself an identifiable entity. Thus, it follows that *Averbuch* also does not teach or suggest generating a profile that includes identifying information for each component file.

Independent claims 17, 21, 22, and 24 are allowable for the reasons stated above with respect to claims 1 and 8. Since claims 23 and 25 depend from claims 22 and 24, respectively, the same distinctions between *Averbuch* and the claimed invention in claims 22 and 24 apply for these claims. Additionally, claims 23 and 25 recite other additional combinations of features not suggested by the reference. Consequently, it is respectfully urged that the rejection of claims 17 and 21-25 is overcome.

Applicant’s claim 31 recites:

31. A data processing system, comprising:
 - a remote control unit; and
 - a base unit connectable to a monitor for providing Internet access under the control of the remote control unit, the base unit comprising:
 - a processor; and
 - a memory electrically connected to the processor, the memory having stored therein Internet protocol instructions and a file transfer download routine to be executed by the processor for performing the following steps:
 - receiving from a server a profile of a download file that includes identifying information for a plurality of component files that make up the download file;
 - initiating a download sequence by which each component file is transferred, one-by-one, from the server using the Internet protocol instructions; and
 - responsive to receipt of the component files for reassembling the component files into the download file using the profile.

Independent claim 31 is allowable for the reasons stated above with respect to claim 1. Furthermore, *Averbuch* does not teach or fairly suggest “a remote control unit” and “a base unit connectable to a monitor for providing Internet access under the control of the remote control unit,” as recited in claim 31. In contradistinction, *Averbuch* teaches a “battery charger/software downloader.” See *Averbuch*, Figure 1, element 108. A “battery charger/software downloader” is not an equivalent to the claimed remote control unit and base unit, because it does not perform the same functions. The “battery charger/software downloader” is not “connectable to a monitor for providing Internet access under the control unit,” as recited in claim 31.

Therefore, the rejection of claims 1-4, 8, 17, 21-25, and 31 under 35 U.S.C. § 102 has been overcome.

II. 35 U.S.C. § 103, Obviousness

The examiner rejects claims 1-31 under 35 U.S.C. § 103 as being unpatentable over Rowe et al. (US Patent No. 5,737,599). The Office Action states that the claims are rejected as stated in prior Office Action (paper #4). This rejection is respectfully traversed.

With respect to claim 1, the Office Action states:

As per claim 1, Rowe teaches a method of downloading a file, consisting of components, from an internet server to a client, comprising the steps of:

generating a profile of the file [col.14 lines 38-60];
initiating a download sequence by which each component is transferred using Internet protocol; and reassembling the components into the file using the profile [col.25 lines 55-68].

Rowe does not specifically disclose downloading one-by-one all of the components prior to reassembling. Rowe discloses downloading only components that needed to display a current page to the user. However, it would have been obvious for one of ordinary skill in the art to download all of the components because it would have enabled the user to have a complete copy of the file and would have enabled the user to use the file off-line.

Office Action, dated 11 August 1999. The Office Action states, “it would have been obvious for one of ordinary skill in the art to download all of the components because it

would have enabled the user to have a complete copy of the file and would have enabled the user to use the file off-line.” Applicant respectfully disagrees. *Rowe* discloses that the prior art downloads the entire document file at one time for use off-line. See col. 2, lines 20-24. The invention described in *Rowe* allows a user to download and view pages of a document file, thus providing an alternative to downloading the entire document file at one time. Thus, if a person of ordinary skill in the art wanted “to have a complete copy of the file” and “to use the file off-line,” the person of ordinary skill would not use the invention of *Rowe*, but rather would use the prior art.

“It is impermissible within the framework of section 103 to pick and choose from any one reference only so much of it as will support a given position, to the exclusion of other parts necessary to the full appreciation of what such reference fairly suggests to one of ordinary skill in the art.” *In re Hedges*, 228 U.S.P.Q. at 687. Thus, when *Rowe* is examined as a whole, *Rowe* teaches one of ordinary skill in the art downloading pages and objects from within a single document file, which teaches away from downloading, one-by-one, component files and reassembling the component files to form an original download file as recited in claim 1. Therefore, one of ordinary skill in the art would not be motivated to make the examiner’s proposed changes.

A *prima facie* case of obviousness cannot be properly based upon a prior art reference if the prior art reference requires some modification in order to meet the claimed invention and such a modification destroys the intended purpose or function of the disclosed invention in the reference.

Furthermore, *Rowe* actually teaches away from the presently claimed invention since *Rowe* directs one to downloading only a portion of a single document file rather than downloading, one-by-one, the component files and reassembling the download file using the identifying information in a profile as in the claimed invention. See *In re Hedges*, 228 U.S.P.Q. 685 (Fed. Cir. 1986). Thus, one of ordinary skill in the art would not be motivated to make the changes proposed by the examiner.

The mere fact that a prior art reference can be readily modified does not make the modification obvious unless the prior art suggested the desirability of the modification. *In re Laskowski*, 871 F.2d 115, 10 U.S.P.Q.2d 1397 (Fed. Cir. 1989) and also see *In re*

Fritch, 972 F.2d 1260, 23 U.S.P.Q.2d 1780 (Fed. Cir. 1992) and *In re Mills*, 916 F.2d 680, 16 U.S.P.Q.2d 1430 (Fed. Cir. 1993). The examiner may not merely state that the modification would have been obvious to one of ordinary skill in the art without pointing out in the prior art a suggestion of the desirability of the proposed modification.

The cited portion of *Rowe* is reproduced as follows:

The process begins at 72. In step 74, an internal list of objects and lists of shared objects are created from the non-optimized document file. These lists help the process to organize the objects of the document for grouping the objects in the more optimized configuration of the present invention. The lists of shared objects are used to place shared objects in the document after the page contents objects. Step 74 is described in greater detail with respect to **FIG. 5**. In next step 76, the document information, including page content information 56, shared objects 60, and special objects 61 are written to the optimized document file according to the internal list organized in step 74. The process essentially retrieves each successive object name or "ID" in the internal list and writes the object to the optimized file in the same order. This has the effect of organizing the page contents 56, shared objects 60, and special objects 61 for all the pages of the document as shown in **FIG. 3b**. In addition, space may be allocated at the beginning of the optimized file to store the cross reference table portion 64 and range table 66, which are described below. Also, information for the cross reference table 58 describing the locations of objects in the file is stored in memory as the objects are written in this step.

Rowe, col. 14, lines 38-60.

In step 228, the first page data and page offset table are downloaded from the source file by the viewer in another connection (or another transaction) using the range table, and the first page is displayed. This first page data includes page contents and shared objects of the first page. The portion of the cross reference table for the first page is also downloaded at this time. The shared objects and page contents of the first page may have already been interleaved using offsets in the range table, as described with reference to **FIG. 9**. Thus, in one embodiment, a portion of the page contents are downloaded, followed by any shared objects referenced by that portion (and cross-reference data for

those shared objects), which allows that portion to be immediately displayed to the user.

Rowe, col. 25, lines 55-68. *Rowe* clearly teaches downloading portions (i.e. pages) from a document file. *Rowe* discloses that shared objects, such as fonts, may be stored in files distinct from the document file. See *Rowe*, col. 2, lines 56-60. However, *Rowe* does not disclose “a file, consisting of a set of component files,” “receiving from the server a profile of the download file that includes identifying information for each component file,” and “reassembling the component files into the download file using the identifying information in the profile,” as recited in amended claim 1. The components of *Rowe* are objects, which are portions of the document file, rather than separately identifiable files. If shared objects are stored as separately identifiable files, they are not reassembled into the original download file. More specifically, *Rowe* does not teach that component files are reassembled into a file using identifying information in a profile.

The present invention recognizes the disadvantage of using file transfer protocol (FTP) for download of large files using a device that is likely to be interrupted during download. *Rowe* does not teach the problem or its source. Instead, *Rowe* is directed towards optimizing a document file so that a page of the document may be efficiently retrieved from the document file and displayed without downloading the entire document file. *Rowe* does not consider a download sequence being interrupted. Therefore, one of ordinary skill in the art would not be motivated to combine or modify the references in the manner required to form the solution disclosed in the claimed invention.

The present invention is directed towards downloading a download file, consisting of a set of component files. Even if *Rowe* could be properly modified as suggested in the Office Action, the modification of *Rowe* would not form the presently claimed invention in claim 1. Instead, the modification of *Rowe* would result in a method of downloading pages of a document file until all of the pages have been downloaded and displayed. The modified teaching of *Rowe* does not result in reassembling the document file “responsive to receipt of the component files,” as recited in amended claim 1.

In addition, the Office Action may not suggest modifications to the prior art using the claimed invention as a model for the modifications. *In re Fritch*, 972 F.2d 1260, 23

U.S.P.Q.2d 1780, 1783-1784 (Fed. Cir. 1992). "The mere fact that the prior art may be modified in the manner suggested by the Examiner does not make the modification obvious unless the prior art has suggested the desirability of the modification." *Id.* In other words, unless some teaching exists in the prior art for the suggested modification, merely asserting that such a modification would be obvious to one of ordinary skill in the art is improper and cannot be used to meet the burden of establishing a *prima facie* case of obviousness. Such reliance is an impermissible use of hindsight with the benefit of applicant's disclosure.

Therefore, absent some teaching, suggestion, or incentive in the prior art, *Rowe* cannot be properly modified to form the claimed invention. As a result, absent any teaching, suggestion, or incentive from the prior art to make the proposed modifications, the presently claimed invention can be reached only through an impermissible use of hindsight with the benefit of applicant's invention as a model.

Since claims 2-7 depend from claim 1, the same distinctions between *Averbuch* and the claimed invention in claim 1 apply for these claims. Additionally, claims 2-7 claim other additional combinations of features not suggested by the reference. Consequently, it is respectfully urged that the rejection of claims 1-7 is overcome.

Particularly, with respect to claim 6, the Office Action states:

As per claim 6, it is well known in the art to use CRC code for identifying file object. A method of producing the unique code would have been a matter of design choice well within the level of one of ordinary skill in the art.

Office Action, dated 11 August 1999. Applicant respectfully disagrees. Cyclical redundancy checking (CRC) codes are typically used for error checking of digitally transmitted data by appending the code onto and sent with a message. The prior art fails to teach or suggest CRC codes in identifying information for each component file of a download file in a profile, which is received before initiating a download of the component files. More specifically, no suggestion to modify *Rowe* to include such a combination of features has been shown in the Office Action. Instead, this combination of features is dismissed as "a matter of design choice well within the level of one of ordinary skill in the art." Thus, the Office Action fails to establish a *prima facie* case of

obviousness.

Further, with respect to claim 7, the Office Action states:

As per claim 7, Rowe does not specifically disclose verifying the component transferred is part of the file. It would have been obvious for one of ordinary skill in the art to do so because it would have improved the reliability of the system.

Office Action, dated 11 August 1999. However, claim 7, as amended, recites “verifying that a component file transferred from the server is part of the download file **using the identifying information.**” Therefore, not only does the Office Action casually dismiss the step of verifying that a component file is part of the download file, but the Office Action fails to address the feature of using the identifying information to verify the component file altogether. Applicant notes that the Office Action neglects to point out any teaching in the prior art that suggests the modification proposed in the rejection. More specifically, the Office Action fails to provide a reference that teaches or suggests “verifying that a component file transferred from the server is part of the download file using the identifying information,” as recited in claim 7. Thus, the Office Action fails to establish a *prima facie* case of obviousness.

Claims 8-10, 12, 14-17, 21-25, and 31 are allowable for the reasons stated above with respect to claims 1-7. Since claims 9, 10, 14-16, 23 and 25 depend from claims 8, 12, 22, and 24, respectively, the same distinctions between *Averbuch* and the claimed invention in claims 8, 12, 22, and 24 apply for these claims. Additionally, claims 8-10, 12, 14-17, 21-25, and 31 recite other additional combinations of features not suggested by the reference. Consequently, it is respectfully urged that the rejection of claims 8-10, 12, 14-17, 21-25, and 31 is overcome.

Particularly, with respect to claim 14, the Office Action states:

As per claims 8-11, 12-16, 17-20, 21, 22-23, they are rejected under similar rationales as for claims 1-7 above.

Office Action, dated 11 August 1999. Claim 14 recites, “re-transferring the profile from the server to the client prior to restarting the download sequence.” This feature is not addressed by the Office Action, particularly with respect to claims 1-7 or claim 14. Thus, the Office Action fails to establish a *prima facie* case of obviousness.

Therefore, the rejection of claims 1-10, 12, 14-17, 21-25, and 31 under 35 U.S.C. § 103 has been overcome.

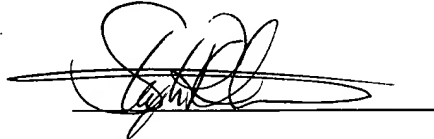
III. Conclusion

It is respectfully urged that the subject application is patentable over *Averbuch* and *Rowe* and is now in condition for allowance.

The examiner is invited to call the undersigned at the below-listed telephone number if in the opinion of the examiner such a telephone conference would expedite or aid the prosecution and examination of this application.

DATE: September 20, 2000

Respectfully submitted,

A handwritten signature in black ink, appearing to read 'S. Tkacs', is written over a horizontal line.

Stephen R. Tkacs
Reg. No. 46,430
Carstens, Yee & Cahoon, LLP
P.O. Box 802334
Dallas, TX 75380
(972) 367-2001